Reusable Templates of Human Performance in Space Shuttle Procedures

Michael Matessa
Human Factors Division
NASA Ames Research Center
Moffett Field, CA, USA
mmatessa@arc.nasa.gov

Roger Remington
Human Factors Division
NASA Ames Research Center
Moffett Field, CA, USA
rremington@arc.nasa.gov

Abstract - *One way to model human behavior easily and accurately is to decompose a complex task into a set of primitive operations to which performance parameters may be assigned. This allows reuse of models at the task level by means of behavioral templates. Performance predictions generated from reusable templates were tested against data from an experiment on space shuttle procedures. The experiment used different participant populations (novice pilots vs. expert astronauts) in different workload conditions (single vs. multiple malfunctions). A phrase-reading template was found to predict performance for the different groups in the different conditions, and a screen-touching template was found to predict performance for astronauts, but there was not enough data to evaluate cross-group and cross-condition predictions. Templates appear to be a useful tool for making predictions of human performance.*

Keywords: *Composable modeling, templates, GOMS*

1   Introduction

One way to model human behavior easily and accurately is to decompose a complex task into a set of primitive operations to which performance parameters may be assigned. These parameters can be static (e.g., 200 msec for a button press) or dynamic (e.g., a Fitts' Law calculation for mouse movement time based on distance and target dimensions). These primitives represent the building blocks from which behavior can be constructed for entire task sequences. This allows reuse of models at the task level by means of behavioral templates.

Model reuse is a profitable avenue to explore for a number of reasons. Task-level skills such as mousing and clicking on a button should be applicable to many HCI tasks, and templates of these skills should not need to be built from scratch for each new project. Previous empirical validation of reused templates should allow for more accurate predictions. Finally, reuse provides additional constraint on models of complex tasks. If the templates predict the behavior well, the HCI modeler should not change the parameters of the template simply to make it work in a new domain.

The GOMS (Goals, Operators, Methods, and Selection rules) [1] modeling methodology has been used to decompose complex tasks into a hierarchical set of nested goals and subgoals. A variant of GOMS called CPM-GOMS [2] creates templates from **C**ognitive, **P**erceptual, and **M**otor operators. CPM-GOMS has been shown to make very accurate *a priori* predictions of human performance in real-world task domains. An example is Project Ernestine, which predicted the outcome of a test of new computer workstations that saved a telephone company $2 million per year [3].

Parameterized templates have been created in CPM-GOMS for commonly recurring task-level activities in HCI, such as mouse moving-and-clicking, or typing, which range from a fraction of a second up to several seconds [4][5]. GOMS has been automated in a computational system, Apex-CPM, that allows the expression of hierarchical goal structure as a nested set of procedures, with the lowest procedures being basic [6][7]. Modeling in this paper utilizes this system.

The task of interest for this paper is fault management in the space shuttle during ascent (the eight and a half minutes of operations from launch to main engine cutoff). Fault management has five sub-phases: being alerted to a fault, identifying the fault, determining the correct procedure for the fault, taking actions to correct the fault, and verifying that the fault has been correctly managed. Most of the time involved in fault management is a result of visually acquiring information, with some time involved in keyboard and switch input actions. Therefore, the modeling has focused on the prediction of gaze durations and motor response time.   Gaze duration is measured by eye tracking. Eye tracking has been used to assess human performance in aviation, but has not yet been used to assess performance in the space shuttle environment.

The Space Shuttle Cockpit Simulator at the Intelligent Spacecraft Interface Systems (ISIS) lab at NASA Ames Research Center permits the collection of eye tracking information and motor responses during shuttle operations. The simulator is reconfigurable for a number of different cockpit designs and uses touch screens to represent the displays, keyboards, and switch panels found in cockpits (see Figure 1). The simulator has been used to obtain data from novice shuttle operators (specially trained airline pilots) and expert operators (current astronauts) in both

```
Keyboard  =  "ACK"                   Procedure  =  "If  2(3)"  +
          =  470  *  1  =  470 msec              "Ps<28.0"  +
                                                 "or>34.0:"  +
Message  =  "MPS  LH2"  +  "/OH2  ULL"           "MPS  LH2"  +
         =  470  *  2  =  940 msec               "ULL PRESS"  +
                                                 "-- OP"  +
Data  =  "25.7↓"  +  "25.6↓"  +  "25.8↓"         "When all"  +
      =  470  *  3  =  1410 msec                 "Ps>34.0:"  +
                                                 "MPS  LH2"  +
Switch  =  "LH2  ULLAGE"+"PRESS"+"OPEN"          "ULL PRESS"  +
        =  470  *  3  =  1410 msec               "-- AUTO"
                                          =  470  *  11  =  5170 msec
```

Table 1: Predictions for reading times

single and multiple malfunction conditions. Since both airline pilots and astronauts are skilled operators of complex flight equipment, it is reasonable to expect that the same templates of skilled phrase reading and motor response should predict performance for both groups and also generalize over conditions of single or multiple malfunctions.

The GOMS methodology was used by Chuah, John, and Pane [8] to predict times for performing tasks using graphic and textual displays. Their model of comprehending visual information from a single fixation is constructed from an attend-target operator lasting 50 msec, an initialize-eye-movement operator lasting 50 msec, an eye-movement operator lasting 30 msec, a perceive-target operator lasting 290 msec, and a verify-target operator lasting 50 msec. Since each operator begins only upon the the completion of the previous operator, the times are additive. This gives a total time of 470 msec per fixation. With their assumption that a fixation can encompass roughly 6 letters in 12-point font, the times for gaze durations during a particular fault in the shuttle environment can be predicted as follows: reading a key on the keyboard requires one fixation for 470 msec, reading a fault message requires two fixations for 940 msec, reading data or a switch label requires three fixations for 1410 msec, and reading a procedure requires eleven fixations for 5170 msec. See Table 1 for details. These predictions were compared to eye movement data collected in the simulator.

The GOMS methodology was also used by Ko [9] to model interactions with touch screens in workstations using a time of 200 msec to predict the motor component of touching the screen. This prediction can be combined with the 470 msec fixation prediction above to create a template for reading a phrase on the screen and touching the screen. The time is 470 msec for one fixation, 50 msec for initializing a screen touch, and 200 msec for executing the touch, for a total of 720 msec. It is assumed that hand movements can be made during reading times. This

prediction was compared to the time between screen touches in the simulator.

1   Experiment

An experiment was run by the ISIS lab to examine the performance of novice and expert shuttle operators in handling single and multiple malfunctions. The experiment was used to test the hypothesis that reusable templates could predict performance of different participant populations in different workload conditions (low workload for single malfunction trials and high workload for multiple malfunction trials).

1.1   Subjects

Five former airlines captains with an average of 15,000 flight hours experience participated in the novice condition. Five current astronauts with a minimum of two years of training participated in the expert condition.

1.1   Equipment

The Space Shuttle Cockpit Simulator at the Intelligent Spacecraft Interface Systems (ISIS) lab at NASA Ames Research Center was used for the experiment. The simulator is a fixed-base, part-task simulator with 4 19" LCD monitors for the 8 front displays, 7 19" touch-screen LCD monitors for the side and overhead switch panels, 1 12" touch-screen LCD monitor for the keyboard, and 2 audio speakers. The setup is shown in Figure 1. The shuttle flight dynamics and system parameter tables were provided by the Shuttle Engineering Simulator at Johnson Space Center. The display graphics were generated with the Virtual Prototypes Incorporated's Visual APplicationS builder (VAPS), a C-based rapid prototyping tool. A head-mounted eye camera (ISCAN ETL-500, ISAN, Inc., Burlington, MA) and head tracker (FasTRAK, Polhemus, Colchester, VT) were used to measure the subjects' eye fixations and movement times.

Figure 1: Setup of the ISIS shuttle simulator.

1.1 Method

Prior to the simulation runs, subjects in the novice condition participated in a 1-week training course covering the following topics: Operational sequence for powered flight; Guidance, navigation, and control; Ascent checklist; and Systems, including Data processing system, Main propulsion system, Environmental control and life support systems, and Auxiliary power units. Subjects in both novice and expert conditions were given a simulator familiarization session.

Each trial simulated the ascent phase of shuttle operations, starting from launch to Main Engine Cut-Off (MECO) and lasting 8 minutes and 30 seconds of simulated mission elapsed time. In the single malfunction condition, a main engine malfunction was inserted. In the multiple malfunction condition, a combination of helium system regulator failure, General Purpose Computer failure, and coolant system failure malfunctions were inserted. The Flight Data File procedure checklist, which lists all the steps required to recover from the malfunctions, was provided to the subjects during the simulation. Eye fixations were analyzed on regions of interest relevant to solving malfunctions.

1.1 Results

Regions of interest in the cockpit were chosen that were relevant to reacting to the malfunctions, and the average duration of gaze time on the regions of interest were measured. The region corresponding to the procedure checklist did not have a visual plane specified by the eye tracker, and so gazes in that region were scored as having "no specific plane". Gazes looking away from the displays in any other direction were also scored as having no specific plane, but an informal review of the data indicated that gazes in no specific plane lasting more than three seconds were always directed to the procedure checklist. This assumption should be verified in future investigations.

For the single malfunction condition, Figure 2 shows the model predictions of gaze durations tested against data from five pilots and four astronauts. One astronaut was excluded from analysis for anticipating the malfunction alarm and solving the malfunction with few eye fixations. Error bars in the figure show the standard error. The average difference of gaze duration between the model and pilots for the five regions of interest was 523 msec. The average difference between the model and astronauts was 520 msec. The average percent error between pilot and model was 22%, and the average percent error between astronaut and model was 29%. The correlation between the gaze durations for the regions of interest of the model and pilots was 0.96. The correlation between the gaze durations of the model and astronauts was 0.99.
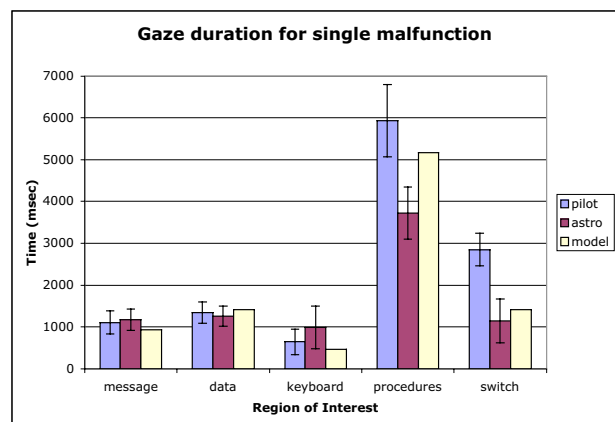

Figure 2: Eye fixation durations of pilots and astronauts in for single malfunction

For the multiple malfunction condition, predicted reading times were calculated as in Table 1, with message, data, keyboard, procedure, and switch areas of interest. The condition included a coolant system failure, which requires reading the data from two systems, so a new region of interest, data2, was added. Only one pilot performed all the necessary actions and one astronaut failed to complete one malfunction solution, so the model was tested only against the four astronauts who solved all of the malfunctions correctly (Figure 3). Error bars in the figure show the standard error. The average difference of fixation time between the model and astronauts for the six regions of interest was 328 msec, and the average percent error was 23%. The correlation between the fixation times for the regions of interest of the model and astronauts was 0.99.

Touch screen timing data from pilots was distorted because some reported difficulties in touching, resulting in multiple rapid touches with times as low as 50 msec. Also, the pilots tended to make decisions and fixations to other locations between screen touches, resulting in between-touch times of over 1200 msec (the time for a GOMS "think" operation).

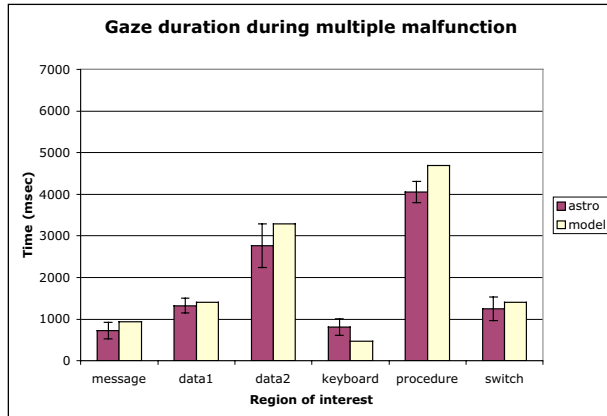**Gaze duration during multiple malfunction**

Figure 3: Eye fixation durations of astronauts for multiple malfunctions

Because of this difficulty, the screen touch template prediction of 720 msec (470 msec fixation + 50 msec initiation + 200 msec execution) was compared to astronaut data for between-touch times less than 1200 msec. Due to the small number of screen touch times, times from both the single and multiple malfunction conditions were combined. The average of these times is 530 msec, which is 190 msec less than the predicted time. A post-hoc analysis of the distribution of times shows two peaks, one around 250 msec and one around 650 msec, with no times between 300 and 400 msec (Figure 4). Of the four times from the single malfunction condition, three were less than 250 msec and one was 438 msec, so the distribution was not due to workload. One explanation for these two sets of times is that since these are times between screen touches, the astronauts might not have needed to re-fixate on the screen. The prediction for this faster strategy would be 50 msec for a touch initation and 200 msec for a touch execution for a total of 250 msec. Looking at the averages of these two sets separately gives averages of 170 msec and 665 msec. This gives a post-hoc 80 msec difference in model prediction for the faster strategy and 55 msec difference for the slower strategy (Figure 5).
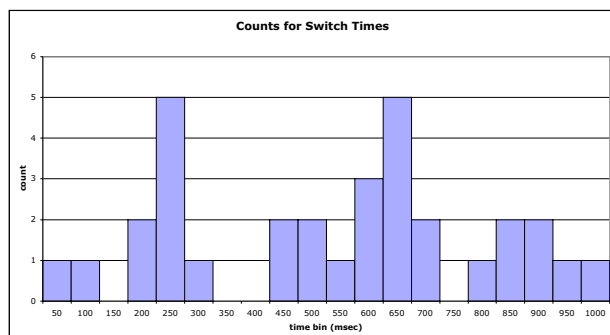

Figure 4: Distribution of screen touch times

1.2 Experiment Conclusions

The experiment tested the hypothesis that reusable templates could successfully predict performance of different participant populations in different experimental conditions. The phrase-reading template made generally good predictions of gaze durations for pilots and astronauts in the single malfunction condition, but there are some noticeable differences. The template represents the process of identifying visual information, but does not include the time to search for the information. The increased gaze time on the switch panel for the pilots relative to the model may represent some search, as the switch panel contains many similar-looking switches (see Figure 6). The template also assumes that all information at a location is processed only once. The decreased gaze time on procedures for the astronauts relative to the template may represent the astronaut reading only a relevant subset of the procedure. Likewise, the increased gaze time on procedures for the pilots relative to the template may be due to the pilots reading parts of the procedure more than once. The phrase-reading template made very good predictions of astronaut gaze duration in the multiple malfunction condition, showing that the template generalizes across workload conditions.

For the screen touch template, there were not enough data points to evaluate generality across population or workload conditions. When creating the screen touch template, it was assumed that eye fixations were made before touching the screen. Close examination of the data suggests this may be incorrect and that some screen touches may be made without fixations.
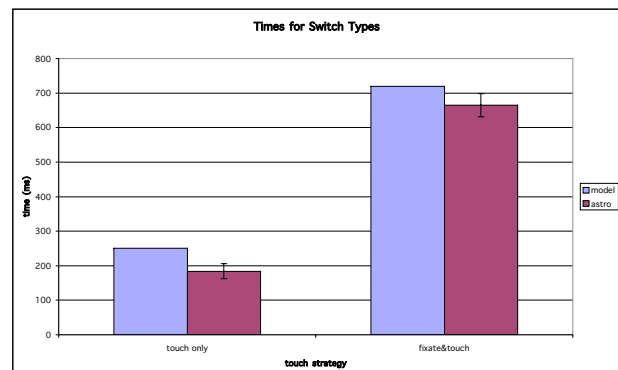

Figure 5: Screen touch times for model and astronauts

2 General Discussion

The phrase-reading and screen-touching templates presented in this paper appear to do a good job of predicting behavior. A post-hoc analysis showed that two screen-touching template strategies, one with reading and one without, accounted for the data better than a single strategy. Also, a more detailed screen touch template could include the time required to move the hand from one

position to another. This may account for more variance in some of the longer screen touch times.

Figure 6: Switchboard layout and close-up

The analyses in this paper include only regions of interest that are relevant to solving malfunctions, but there is also a parallel task of monitoring the entire state of the shuttle. Future research will be directed towards modeling the multitasking behavior of the shuttle operators as they solve malfunctions while at the same time monitor the overall state of the shuttle.

The next step is to chain these templates into longer sequences of behavior. Some preliminary work has been done on this front [10], and the data show that subject sequencing strategies are highly variable both between novice/expert groups and within these groups. The Apex-CPM system has been useful in representing the probabilistic nature of transitions between templates, but determining subject strategy in order to make predictions will be a new challenge.

3 Acknowledgements

References

[1] Card, S. K., Moran, T.P. & Newell, A. (1983). The Psychology of Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum Associates.

[2] John, B. E. (1990) Extensions of GOMS analyses to expert performance requiring perception of dynamic visual and auditory information. In Proceedings of CHI, 1990 (Seattle, Washington, April 30-May 4, 1990) ACM, New York, 107-115.

[3] Gray, W. D., John, B. E. & Atwood, M. E. (1993) Project Ernestine: Validating a GOMS Analysis for Predicting and Explaining Real-World Task Performance, Human-Computer Interaction, v.8 (3), pp.237-309.

[4] John, B. E. & Gray, W. D. (1992) GOMS Analyses for Parallel Activities. Tutorial materials, presented at CHI, 1992 (Monterey, California, May 3- May 7, 1992), CHI, 1994 (Boston MA, April 24-28, 1994) and CHI, 1995 (Denver CO, May 7-11, 1995) ACM, New York.

[5] Gray, W. D., & Boehm-Davis, D. A. (2000). Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. Journal of Experimental Psychology: Applied, 6(4), 322-335.

[6] John, B. E., Vera, A. H., Matessa, M., Freed, M., and Remington, R. (2002) Automating CPM-GOMS. In Proceedings of CHI'02: Conference on Human Factors in Computing Systems. ACM, New York.

[7] Vera, A., John, B., Remington, R., Matessa, M., & Freed, M. (in press). Automating Human-Performance Modeling at the Millisecond Level. Human-Computer Interaction.

[8] M.C. Chuah, B.E. John, and J. Pane (1994) "Analyzing Graphic and Textual Layouts with GOMS: Results of a Preliminary Analysis," CHI 94 Conference Companion: Conference on Human Factors in Computing Systems, Boston, MA, April 1994, pp. 323-324.

[9] H. Ko. (2000) "Open Systems Advanced Workstation Transition Report" SSC San Diego TR-1822, July 2000.

[10] Matessa, M. & Remington, R. (2005). Eye Movements in Human Performance Modeling of Space Shuttle Operations. Human Factors and Ergonomics Society Conference. Orlando, FL.